



Relatório Técnico

**Núcleo de
Computação Eletrônica**

A Validity Measure for Hard and Fuzzy Clustering derived from Fisher's Linear Discriminant

**C. R. de Franco
L. S. Vidal
A. J. de O. Cruz**

NCE - 02/02

Universidade Federal do Rio de Janeiro

A Validity Measure for Hard and Fuzzy Clustering derived from Fisher's Linear Discriminant

Cláudia Rita de Franco

Leonardo Silva Vidal

Adriano Joaquim de Oliveira Cruz

Universidade Federal do Rio de Janeiro – AEP/NCE

Caixa Postal 2324 – Ilha do Fundão – CEP. 20001-970 – Rio de Janeiro, RJ, Brasil

Abstract – Cluster analysis has a growing importance in many research areas, especially those involving problems of pattern recognition. Generally, in real world problems, the number of classes is unknown in advance, being necessary to have criterions to identify the best choice of clusters. Here we propose an extension to Fisher Linear Discriminant, the EFLD that does not impose limits on the minimum number of samples, can be applied to fuzzy and crisp partitions and can be calculated more efficiently. We also propose a new fast and efficient validity method based in the EFLD that measures the compactness and separation of partitions produced by any fuzzy or crisp clustering algorithm. The simulations performed indicate that it's a efficient and fast measure even when the overlapping between clusters is high. Finally, we propose an algorithm that applies the new validity measure to the problem of finding the patterns for the fuzzy K-NN classifier. This algorithm is applied to the problem of cursive digits recognition.

Key words: Cluster validity, fuzzy clustering, pattern recognition, cursive digits recognition, separate and compact clusters, Fisher's Linear Discriminant.

I. INTRODUCTION

Clustering techniques are used to partition data sets in subsets or clusters that show a certain degree of closeness or similarity. Hard partitions assign each element of the data set to one and only one cluster assuming well-defined boundaries among clusters. Very often, these boundaries are not so well defined and this kind of partition does not describe the underlying data structure. Thus, numerous problems are best solved by fuzzy partitions where each element may belong to various clusters with different membership degrees. In fuzzy clustering, the membership degrees are real values between 0 and 1 and in hard clustering, the membership degree is equal to one for the samples belonging to the cluster and zero for the others.

There are some difficulties when clustering real data. The number of clusters, very often, is unknown a priori and the distribution of points among clusters is influenced by the clustering algorithm and may be not optimal. Therefore, it is important to find a criterion to determine the best number of clusters that represents the data set and the quality of the clustering result. Several validity measures have been proposed to validate this by calculating the relative compactness of each cluster and the separation among all clusters for a given partition set.

The Partition Coefficient F and the Partition Entropy Coefficient H can be used to find the optimum number of

fuzzy partitions [1]. However, these indices are influenced by the degree of overlapping between clusters and their efficiency decreases in these situations. The Minimum and Maximum Relative Fuzziness index measure the degree of separation among fuzzy clusters, so it can be used to validate the quality of a clustering process after the number of clusters was determined [2]. This index also suffers as the superposition among clusters increases. The function S is a more complete fuzzy measure since it evaluates the quality of the clustering process as well as the number of clusters [3]. The Fisher's Linear Discriminant evaluates the compactness and separation of hard partitions and is usually applied in problems of pattern recognition [4].

This article is organized as follows. In Section II, we propose an extended Fisher's Linear Discriminant that can be applied to fuzzy and hard partitions. In Section III, we propose a new validity measure that can also be applied to fuzzy and hard partitions. In Section IV, we present numerical justifications for both validity measures. In Section V, we describe an application of our validity measure to the problem of cursive digits recognition.

II. EXTENDED FISHER LINEAR DISCRIMINANT

The Fisher's Linear Discriminant (FLD) is an important technique used in pattern recognition problems to evaluate the compactness and separation of the partitions produced by crisp clustering techniques.

The scatter matrices used by FLD can be applied only to hard partitions. We are proposing extended versions of these matrices that can be applied to fuzzy and crisp partitions.

The extended between-class scatter matrix (1) estimates the number of points in a cluster as the sum of all point's fuzzy memberships on the cluster. The extended within-class scatter matrix (2) evaluates each cluster's scattering as a function of all points' scattering and their fuzzy memberships in the cluster. Both became the Fisher's scatter matrices if all partitions are hard.

$$S_{Be} = \sum_{i=1}^c \sum_{j=1}^n \mu_{ij} \cdot (\mathbf{m}_i - \mathbf{m}) \cdot (\mathbf{m}_i - \mathbf{m})^T \quad (1)$$

$$S_{We} = \sum_{i=1}^c \sum_{j=1}^n \mu_{ij} \cdot (\mathbf{x}_j - \mathbf{m}_i) \cdot (\mathbf{x}_j - \mathbf{m}_i)^T \quad (2)$$

where the centroid of the i^{th} partition is given by

$$\mathbf{m}_{ei} = \frac{\sum_{j=1}^n \mu_{ij} \cdot \mathbf{x}_j}{\sum_{j=1}^n \mu_{ij}} \quad (3)$$

and the centroid of the whole data set is

$$\mathbf{m} = \frac{1}{n} \cdot \sum_{j=1}^n \mathbf{x}_j \quad (4)$$

where n is the number of data points, c is the number of clusters, \mathbf{x}_j ($j=1,2,\dots,n$) is a column vector representing the j^{th} data point and μ_{ij} is the fuzzy membership of the j^{th} data point in the i^{th} cluster.

If the fuzzy memberships follow (5), the sum of S_{Be} and S_{We} is equal to Fisher's total scatter matrix (6) as shown in Appendix A.

$$\forall j, \sum_{i=1}^c \mu_{ij} = 1 \quad (5)$$

$$S_T = \sum_{j=1}^n (\mathbf{x}_j - \mathbf{m}) \cdot (\mathbf{x}_j - \mathbf{m})^T \quad (6)$$

The criterion function J_e of the extended FLD (EFLD) is shown in (7). The Partition sets with better compactness and separation are characterized by higher values of J_e .

$$J_e = \frac{|S_{Be}|}{|S_{We}|} \quad (7)$$

The evaluation of the determinants imposes limits on the minimum number of points on each partition. Fukunaga proposed an alternative criterion function for FLD that uses the trace of the scatter matrices [5]. Its extended version J_e is shown in (8).

$$J_e = \frac{\text{trace}(S_{Be})}{\text{trace}(S_{We})} \quad (8)$$

This form of criterion function is a good index to evaluate compactness and separation. It is possible to improve the time to evaluate equation (8) if we observe that the trace of a matrix, produced by the product of a column vector and its transpose is equal to the square of the module of this vector. Therefore, the traces of the matrices on (1) and (2) are given by equations (9) and (10).

$$S_{Be} = \text{trace}(S_{Be}) = \sum_{i=1}^c \sum_{j=1}^n \mu_{ij} \cdot \|\mathbf{m}_{ei} - \mathbf{m}\|^2 \quad (9)$$

$$S_{We} = \text{trace}(S_{We}) = \sum_{i=1}^c \sum_{j=1}^n \mu_{ij} \cdot \|\mathbf{x}_j - \mathbf{m}_{ei}\|^2 \quad (10)$$

Usually, the EFLD should be used to measure relative compactness and separation of different partition sets applied on the same data set. Its evaluation can be further optimized by observing that the sum of the traces of S_{We} and S_{Be} is constant for a given data set and it is equal to the trace of S_T shown in equation (11).

$$S_T = \text{trace}(S_T) = \sum_{j=1}^n \|\mathbf{x}_j - \mathbf{m}\|^2 \quad (11)$$

Thus, the extended criterion function J_e can be rewritten in the form of equation (12), which is faster to evaluate. The term S_T is calculated only once for the data set and S_{Be} is calculated for each partition set and as we can see from (9) and (10), S_{Be} is much faster to evaluate than S_{We} .

$$J_e = \frac{S_{Be}}{S_T - S_{Be}} \quad (12)$$

III. A NEW VALIDITY METHOD

In this section, we define a new index to validate partition's compactness and separation.

The EFLD, that we proposed in Section II, like its traditional version, tend to produce increasingly higher values as the number of partitions rises, as will be shown in Section IV. This tendency grows further worse on data sets with high overlapping among the classes. This problem arises because the clustering algorithms, either fuzzy or hard, have to associate more than one center to each class when the number of clusters c is greater than the number of classes. Its consequence is the decreasing of S_{We} and the increasing of J_e .

When two or more clusters span the same class, the distance between their centers is usually smaller than the distance between centers spanning different classes. This situation can be identified by a sharp decrease of D_{min} (13), which corresponds to the minimum Euclidian distance between all pairs of centers.

$$D_{min} = \min_{1 \leq i \leq c} \left[\min_{i+1 \leq j \leq c-1} \|\mathbf{m}_{ei} - \mathbf{m}_{ej}\| \right] \quad (13)$$

The association of EFLD and D_{min} allows the proposition of a new validity measure for crisp and fuzzy partitions called *Inter Class Contrast* (ICC), shown in (14), which stops increasing when the number of clusters is greater than the number of classes.

$$ICC = \frac{S_{Be}}{n} \cdot D_{min} \cdot \sqrt{c} \quad (14)$$

The term S_{Be} in (14) is an approximation to the EFLD with the same behavior and faster to estimate. It estimates the quality of the placement of the centers on their clusters. A misplaced center would produce small values for S_{Be} .

The values obtained using (8) and (12) are obviously equal, but (12) is significantly faster. For the well separated data set X1, the EFLD correctly identifies five as the number of clusters. Its tendency to grow with c when the classes' overlap is high becomes apparent for data set X2 as it identifies ten as the best number of clusters.

C. Evaluation of ICC

The ICC was applied to each output of FCM and was compared with the partition coefficient F and the validity function S . The maximum value of F and ICC and the minimum value of S indicate their choice for the best number of clusters and disposition of centers.

TABLE III
RESULTS OF F, S AND ICC TO DATA SET X1

No of Clusters	F		S		ICC	
	X1	Time(s)	X1	Time(s)	X1	Time(s)
2	0.70	0.0048	0.35	0.027	7.6	0.0055
3	0.71	0.0052	0.09	0.033	41.9	0.0072
4	0.79	0.0053	0.07	0.037	51.9	0.0083
5	0.94*	0.0063	0.01*	0.047	96.7*	0.0093
6	0.87	0.0059	0.73	0.066	8.72	0.0095
7	0.80	0.0065	0.80	0.068	8.07	0.0098
8	0.75	0.0072	0.69	0.064	8.77	0.0107
9	0.11	0.0074	22285	0.073	0.0002	0.0113
10	0.71	0.0080	0.58	0.075	10.86	0.0124

TABLE IV
RESULTS OF F, S AND ICC TO DATA SET X2

No of Clusters	F		S		ICC	
	X2	Time(s)	X2	Time(s)	X2	Time(s)
2	0.75*	0.0044	0.165	0.021	5.05	0.0057
3	0.62	0.0050	0.224	0.031	4.94	0.0083
4	0.59	0.0049	0.19	0.042	6.2	0.0094
5	0.58	0.0057	0.122*	0.048	7.83*	0.0107
6	0.53	0.0058	0.224	0.054	6.49	0.0118
7	0.49	0.0060	0.216	0.066	6.15	0.0126
8	0.47	0.0061	0.227	0.095	6.08	0.0115
9	0.45	0.0074	0.217	0.149	6.21	0.0121
10	0.43	0.0080	0.223	0.092	5.69	0.0148

Tables III and IV show the values and the execution times in seconds obtained for each partition set on data sets X1 and X2, respectively. To the well-separated data set X1, the measures F , S and ICC validated properly the best number of

clusters as five. To the data set X2, F shows its natural decreasing trend as a function of c by choosing two clusters. In contrast, ICC and S evaluated the right number of clusters as five. F is the faster measure but it failed validating the best number of clusters for X2. The proposed measure ICC is consistently faster than S .

V. APPLICATION OF ICC TO THE CHOICE OF PATTERNS FOR FUZZY K-NN

In this section, we propose a Non-Parametric Statistical Pattern Recognition System that associates FCM, ICC and fuzzy K-NN classifier [7] [8]. We also evaluated this system in comparison with the fuzzy clustering methods FCM, Gath-Geva (GG), Gustafson-Kessel (GK) and fuzzy K-NN with randomly chosen patterns in the problem of cursive digits recognition [9]-[10].

A. Description of the system

The fuzzy K-NN classifier is a well-known and high performance fuzzy classification method. The key to its successful performance is the selection of a high-quality set of patterns to represent each class. The FCM is fast and efficient in finding raw sample concentrations and does not impose limits on the minimum number of points like GG and GK do.

Associating the advantages of these two fuzzy methods, we propose the ICC-KNN, a Non-Parametric Statistical Pattern Recognition System that uses FCM and ICC to find the best patterns of a data set and evaluates the best number of neighbor and weight exponent to be used by the fuzzy K-NN. In the Design (training) Phase (DP), the ICC-KNN partitions separately each class of the design data set using the FCM for a range of number of clusters and validates the centers disposition using ICC. The centers that attain the highest ICC value for each class are chosen to be the patterns of that class. Then, the fuzzy K-NN classifier is applied on the whole design data set using the chosen patterns for a range of values of the weight exponent m and the number of neighbors K . This is done in order to determine the configuration with maximum success rate, i.e., the number of samples that had maximum membership on its real class divided by the number of design samples.

In the Test Phase (TP), the ICC-KNN applies the fuzzy K-NN with the parameters from the DP in the test data. Let $X = \{x_1, \dots, x_n\} \in \mathbb{R}^p$ be a set of n labeled samples, the algorithm is as follows:

Algorithm of ICC-KNN

DESIGN PHASE

BEGIN

Set the weight exponent m and the minimum and maximum number of clusters c (c_{\min} and c_{\max})

FOR EACH class s

FOR $c = c_{\min}$ TO c_{\max}

Apply FCM to the points of s using c and m

The square root of c in (14) prevents ICC from reaching its maximum value for a c smaller than the optimum. This would occur when one or more clusters span more than one class since their centers are very far from each other, yielding high values for D_{min} and ICC. The square root of c forces ICC to grow with the number of clusters, thus reaching its maximum values closer to the optimum c while D_{min} avoids the maximum value for a c bigger than the optimal value.

The factor $1/n$ in (14) is a scaling factor to compensate the influence of the number of points on the s_{Be} .

IV. NUMERICAL JUSTIFICATIONS

In this section, we show the behavior of EFLD using the criterion functions defined in Section II and the behavior of the proposed measure ICC against two well know validity measures, F and S, when used to validate partition sets' compactness and separation.

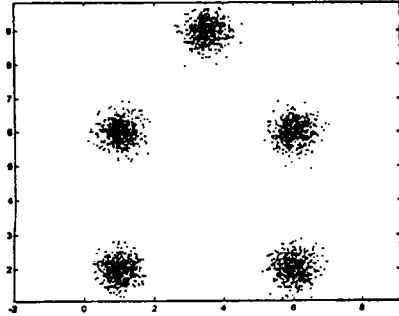


Figure 1. Data set X1

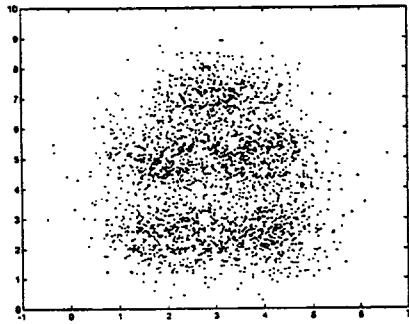


Figure 2. Data set X2

with standard deviation 0.7. As we can see in Figure 1, the five classes in data set X1 are well separated, offering no difficulty to clustering algorithms. In Figure 2, the five classes in data set X2 can not be easily identified due to the high overlapping of the classes. The FCM algorithm was applied to both data sets for $c=[2,...,10]$ and for the exponent weight $m=2$ [6].

B. Evaluation of EFLD

The EFLD was applied to each output of FCM using J_c with the determinant (7) and the trace (8) of the scatter matrices, and with the criterion function proposed in (12).

TABLE I
RESULTS OF EFLD TO DATA SET X1

No of Clusters	EFLD with (7) X1		EFLD with (8) X1		EFLD with (12) X1	
		Time(s)		Time(s)		Time(s)
2	0	0.88	0.18	0.69	0.18	0.0053
3	0.95	1.21	0.98	1.04	0.98	0.0071
4	3.96	1.77	1.87	1.38	1.87	0.0063
5	182*	2.03	13.6*	1.72	13.6*	0.0080
6	164	2.21	12.9	2.06	12.9	0.0093
7	157	2.69	12.7	2.4	12.7	0.0113
8	165	2.75	12.9	2.8	12.9	0.0107
9	0	3.5	0.001	3.1	0.001	0.0118
10	135	4.13	11.7	3.59	11.7	0.0121

TABLE II
RESULTS OF EFLD TO DATA SET X2

No of Clusters	EFLD with (7) X2		EFLD with (8) X2		EFLD with (12) X2	
		Time(s)		Time(s)		Time(s)
2	0	0.82	0.45	0.77	0.45	0.0063
3	0.04	1.26	0.58	1.07	0.58	0.0088
4	0.31	1.68	0.83	1.42	0.84	0.0096
5	0.74	2.12	1.09	1.77	1.09	0.0110
6	0.75	2.5	1.10	2.12	1.10	0.0096
7	0.88	3.19	1.18	2.71	1.18	0.0113
8	1.01	3.29	1.23	3.19	1.23	0.0116
9	1.09	3.45	1.29	3.35	1.29	0.0126
10	1.2*	3.65	1.34*	3.50	1.34*	0.0127

A. Data sets descriptions

Two data sets were artificially produced by generating 500 random points for each class. The classes on data set X1 were centered at the points (1, 2), (6, 2), (3.5, 9), (1, 6), (6, 6) with standard deviation 0.3. In data set X2, the classes were centered at points (2, 2.5), (4, 2.5), (3, 7), (2, 5) and (4, 5)

The maximum value of EFLD indicates the best number of clusters in all cases. Tables I and II show the values and the execution times in seconds obtained for each partition set on data sets X1 and X2, respectively.

```

    Evaluate ICC to the FCM fuzzy memberships
  END FOR
  Determine the centers of the FCM output with
  maximum ICC as patterns for  $s$ 
END FOR EACH
Set the minimum and maximum weight exponent  $m$ 
( $m_{\min}$ ,  $m_{\max}$ ) and number of neighbors  $K$  ( $K_{\min}$ ,  $K_{\max}$ )
FOR  $m = m_{\min}$  TO  $m_{\max}$ 
  FOR  $K = K_{\min}$  TO  $K_{\max}$ 
    Evaluate fuzzy K-NN for the points of  $X$  and
    the patterns chosen by ICC
    Initialize  $i = 0$ 
    FOR EACH point  $x_j$  in the data set
      IF  $x_j$ 's higher membership is in its class
      THEN increment  $i$ 
    END FOR EACH
  END FOR
END FOR
Set  $m$  and  $K$  for K-NN with maximum  $i$ 
IF (a tie exists) choose the smaller  $K$  and  $m$  in the tie
END

```

TEST PHASE BEGIN

Evaluate the fuzzy K-NN for the test data using the patterns, m and K from the DESIGN PHASE
END

B. Application to Cursive Digit Recognition

The ICC-KNN was compared to fuzzy K-NN using random patterns, FCM, GK and GG using a data set of cursive digits codified as square 128 [11]. In order to avoid singular fuzzy covariance matrices on GG and GK, PCA was used to reduce the number of features from 128 to 19, preserving 82.6% of the total variance [12]. The 80% of the samples on each class were used for the DP and the remaining 20% were used for the TP.

In the DP of ICC-KNN, the FCM partitioned each of the ten classes of the problem, one for each digit, using the weight exponent $m=1.25$ and the number of the clusters c varying from 2 to 30. The best numbers of patterns for each class validated by ICC were, respectively, 22, 29, 12, 25, 15, 26, 25, 23, 10 and 30. The fuzzy K-NN classifier was evaluated with the patterns chosen by ICC, K varying from 3 to 7 and $m \in \{1.1, 1.25, 1.5, 2\}$. The configuration with the higher success rate was obtained using $m=1.25$ and $K=6$. In the TP, these parameters were used by the fuzzy K-NN with the chosen patterns on the test data set. The fuzzy K-NN was also applied to the test data set with $K=6$, $m=1.25$ and the patterns chosen randomly from the training data set. The number of patterns for each class was the same as those obtained by ICC.

In order to perform the comparison, the fuzzy clustering methods FCM, GG and GK were used in three simple classification systems with one version for each method. Each system also comprises a Design and a Test Phase. In the

DP, the clustering method was applied on the whole training set in order to identify ten clusters, one for each class, using $m=1.25$. Each cluster was then associated to the class whose points produced the higher sum of memberships on the given cluster. On the TP, the step of the method that evaluates the membership degrees was executed for the test set's points using the centers and the metrics produced on the DP. The success rate of each system was evaluated dividing the number of samples that had their maximum membership on the cluster associated to their real class by the number of test samples.

TABLE V
RESULTS OF K-NN USING ICC-KNN PATTERNS, K-NN USING
RANDOM PATTERNS, FCM, GG AND GK

	ICC-KNN	Random K-NN	FCM	GG	GK
Success Rate	86.7%	75.22%	57%	51%	49%
Execution Time (in seconds)	1784	260	30.38	108.15	711.77

As Table V shows, the ICC-KNN obtained the best result, which compensates the greater execution time. The low performance of FCM, GG and GK is consequence of clusters containing points of different classes. These results show that the ICC-KNN is able to find patterns that represent the data's structure more efficiently.

VI. CONCLUSIONS

We extended and optimized the Fisher's Linear Discriminant so that it can be applied to fuzzy and crisp partitions. It showed good accuracy and faster execution times, failing as it was expected when the overlapping among classes was high. This problem was solved by the proposed index (ICC) that evaluated correctly the number of clusters even when the overlapping among clusters is high. The simulation showed better precision than F and better execution times than S with similar precision.

We also proposed a new classification system (ICC-KNN) that integrates the new index ICC, the FCM and the fuzzy K-NN classifier. The ICC-KNN obtained the highest success rate on the cursive digit classification problem, showing to be more efficient than the fuzzy K-NN using random patterns and than the systems that use other fuzzy clustering methods.

ACKNOWLEDGMENT

This work was supported in part by NCE – Núcleo de Computação Eletrônica, FAPERJ – Fundação de Amparo à Pesquisa do Estado do Rio de Janeiro and CAPES – Fundação de Aperfeiçoamento de Pessoal de Nível Superior.

REFERENCES

- [1] J.C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum Press, New York, 1981
- [2] H.L. Gordon and R.L. Somorjai, "Fuzzy Cluster Analysis of Molecular Dynamics Trajectories," *PROTEINS: Structure, Function and Genetics*, v. 14, p. 249-264, 1992
- [3] X.L. Xie and G.A. Beni, "A Validity Measure for Fuzzy Clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 13, n. 8, August 1991
- [4] Christopher M. Bishop, Neural Networks for Pattern Recognition. Oxford University Press, 1995
- [5] K. Fukunaga, Introduction to Statistical Pattern Recognition, Second ed., San Diego: Academic Press, 1990
- [6] Timothy J. Ross, Fuzzy logic with engineering applications, McGraw-Hill International Editions, Electrical Engineering Series, 1997
- [7] S. Horikawa, "Fuzzy Classification System using Self-Organizing Feature Map," *Oki Technical Review*, v. 63, n. 159, July 1997
- [8] J.M. Keller, M.R. Gray and J.A. Jr. Givens, "A Fuzzy K-Nearest Neighbor Algorithm," *IEEE Transaction on Systems, Man and Cybernetics*, v. SMC-15, n. 4, p. 580-585, July/August 1985
- [9] I. Gath and A.B. Geva, "Unsupervised optimal fuzzy clustering," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, v. 11, n. 7, p. 773-781, July 1989
- [10] D.E. Gustafson and W.C. Kessel, "Fuzzy clustering with a fuzzy covariance matrix," *IEEE Conference on Decision and Control*, p. 761-766, January 1979
- [11] R. J. Rodrigues, E. Silva and A.C.G. Thomé, "Feature Extraction Using Contour Projection," *The 5th World Multi-Conference on Systemics*, Florida, July 2001
- [12] R. Johnson and D. Wichern, Applied Multivariate Statistical Analysis, Prentice Hall International, Inc, 1992

APPENDIX A

If the partition set obeys (5), the sum of the extended within-class scatter matrix and the extended between-class scatter matrix equals Fisher's total scatter matrix.

Proof:

From (3), we have

$$m_{ei} \cdot \sum_{j=1}^n \mu_{ij} = \sum_{j=1}^n \mu_{ij} \cdot x_j \quad (A1)$$

Expanding (2)

$$S_{we} = \sum_{i=1}^c \sum_{j=1}^n \mu_{ij} \cdot (x_j \cdot x_j^T - x_j \cdot m_{ei}^T - m_{ei} \cdot x_j^T + m_{ei} \cdot m_{ei}^T)$$

Reordering

$$S_{we} = \sum_{i=1}^c \left[\sum_{j=1}^n \mu_{ij} \cdot x_j \cdot x_j^T - \left(\sum_{j=1}^n \mu_{ij} \cdot x_j \right) \cdot m_{ei}^T - m_{ei} \cdot \left(\sum_{j=1}^n \mu_{ij} \cdot x_j \right)^T + \left(\sum_{j=1}^n \mu_{ij} \right) \cdot m_{ei} \cdot m_{ei}^T \right]$$

Using (A1) on the second and third terms

$$S_{we} = \sum_{i=1}^c \left[\sum_{j=1}^n \mu_{ij} \cdot x_j \cdot x_j^T - \left(\sum_{j=1}^n \mu_{ij} \right) \cdot m_{ei} \cdot m_{ei}^T - \left(\sum_{j=1}^n \mu_{ij} \right) \cdot m_{ei} \cdot m_{ei}^T + \left(\sum_{j=1}^n \mu_{ij} \right) \cdot m_{ei} \cdot m_{ei}^T \right]$$

And follows that

$$S_{we} = \sum_{i=1}^c \left[\sum_{j=1}^n \mu_{ij} \cdot x_j \cdot x_j^T - \left(\sum_{j=1}^n \mu_{ij} \right) \cdot m_{ei} \cdot m_{ei}^T \right] \quad (A2)$$

Expanding (1)

$$S_{be} = \sum_{i=1}^c \left[\left(\sum_{j=1}^n \mu_{ij} \right) \cdot (m_{ei} \cdot m_{ei}^T - m_{ei} \cdot m^T - m \cdot m_{ei}^T + m \cdot m^T) \right]$$

Reordering

$$S_{be} = \sum_{i=1}^c \left[\left(\sum_{j=1}^n \mu_{ij} \right) \cdot m_{ei} \cdot m_{ei}^T - \left(\sum_{j=1}^n \mu_{ij} \right) \cdot m_{ei} \cdot m^T - \left(\sum_{j=1}^n \mu_{ij} \right) \cdot m \cdot m_{ei}^T + \left(\sum_{j=1}^n \mu_{ij} \right) \cdot m \cdot m^T \right]$$

Using (A1) on the second and third items

$$S_{be} = \sum_{i=1}^c \left[\left(\sum_{j=1}^n \mu_{ij} \right) \cdot m_{ei} \cdot m_{ei}^T - \left(\sum_{j=1}^n \mu_{ij} \cdot x_j \right) \cdot m^T - m \cdot \left(\sum_{j=1}^n \mu_{ij} \cdot x_j \right)^T + \left(\sum_{j=1}^n \mu_{ij} \right) \cdot m \cdot m^T \right] \quad (A3)$$

Adding (A2) and (A3)

$$S_{we} + S_{be} = \sum_{i=1}^c \left[\sum_{j=1}^n \mu_{ij} \cdot x_j \cdot x_j^T - \left(\sum_{j=1}^n \mu_{ij} \cdot x_j \right) \cdot m^T - m \cdot \left(\sum_{j=1}^n \mu_{ij} \cdot x_j \right) + \left(\sum_{j=1}^n \mu_{ij} \right) \cdot m \cdot m^T \right]$$

Putting all the sums in evidence

$$S_{we} + S_{be} = \sum_{i=1}^c \sum_{j=1}^n \mu_{ij} \cdot (x_j \cdot x_j^T - x_j \cdot m^T - m \cdot x_j^T + m \cdot m^T)$$

Regrouping

$$S_{we} + S_{be} = \sum_{j=1}^n \left(\sum_{i=1}^c \mu_{ij} \right) \cdot (x_j - m) \cdot (x_j - m)^T$$

And by (5), we have

$$S_{we} + S_{be} = \sum_{j=1}^n (x_j - m) \cdot (x_j - m)^T = S_T$$